

# JS-SDK 使用说明

## 1 引入文件

页面中引入以下js以及css文件

```
<link rel="stylesheet" href="./static/css/bootstrap.min.css">
<link rel="stylesheet" href="./static/css/jquery-confirm.min.css" >
<link rel="stylesheet" href="./static/css/webapp.css?v=7896112516">

<script type="text/javascript" src="./static/js/jquery1.10.min.js"></script>
<script type="text/javascript" src="./static/js/bootstrap.min.js"></script>
<script type="text/javascript" src="./static/js/jquery-confirm.min.js">
</script>
<script type="text/javascript" src="./static/js/jquery.qrcode.min.js">
</script>
<script type="text/javascript" src="./static/js/custom-protocol-
check.min.js"></script>
<script type="text/javascript" src="./static/js/adaptor-8.0.js"></script>
<script type="text/javascript" src="./static/js/webstream.min.js?
v=1.312717c741"></script>
<script type="text/javascript" src="./static/js/webaudio.min.js?v=20223330">
</script>
<script type="text/javascript" src="./static/js/dolittlecloud.pack.min.js?
v=1.369169"></script>
<script type="text/javascript" src="./static/js/vconsole.min.js?v=980666">
</script>
<script type="text/javascript" src="./static/js/md5.js"></script>
```

## 2 定义渲染元素

html的body中添加id为desktop的div元素

```
```html
<body>
  <div id="desktop"></div>
</body>
```
```

## 3 调用sdk示例

### 调用示例

```
var webServerAddr= "http://127.0.0.1:9000/";
//下面三个参数不建议在前端生成，而是由后端生成，渲染到前端，或者ajax异步从后端获取，
保证userSign的生成规则不暴露在前端
var reqTime = Math.ceil(new Date().getTime()/1000).toString(); //当前时间戳
var userKey = "153d6d17"; //请求key
```

```
var userSign= getUserSign(reqTime,userKey); //请求Sign,生成算法 userSign =  
Md5(Md5(userKey+"doliticlouddesktop")+Md5(userKey+reqTime))
```

```
var LoadParams = {  
    webServerAddr : webServerAddr,  
    reqTime       : reqTime,  
    userKey        : userKey,  
    userSign       : userSign,  
    connToken      : "306c5463989b048354d6928939c15f1e",  
    connMode       : "sdk",  
    runType        : "app",  
    pointerMode    : "default",  
    moveMode       : "default",  
    touchMode      : "default",  
    loadSkin       : "default",  
    loadType       : "stream",  
    group          : "",  
    forceRenderMode : "default",  
    isForceControlMode: false,  
    isShowViewOnlyMask: true,  
    isShowSetBar    : true ,  
    isShowHandleBar : true ,  
    isDefaultShowHandleBar : true,  
    isForceHorizontal : true,  
    isShowKeyBar    : true ,  
    isShowMenu      : false ,  
    isShowParam     : false ,  
    isShortcutParam: true ,  
    isShowPointer   : true ,  
    isShowRemotePointer : false ,  
    isReceiveAudio : true ,  
    isAutoExists    : 1 ,  
    isShowReloadMenu : true ,  
    openMicrophone : false, //开启麦克风, 必须https方式  
    joystickLeftKey : cloudDesktopApp.keyMsgType.KEYCODE_LEFT_ARROW,  
    joystickRightKey : cloudDesktopApp.keyMsgType.KEYCODE_RIGHT_ARROW,  
    joystickDownKey : cloudDesktopApp.keyMsgType.KEYCODE_DOWN_ARROW,  
    joystickUpKey : cloudDesktopApp.keyMsgType.KEYCODE_UP_ARROW,  
    ignoreKeyboardKeys : [  
        cloudDesktopApp.keyMsgType.KEYCODE_F11  
    ],  
};  
var dataChannelConnected = false;  
//应用画面将显示在id为desktop对应的dom元素上  
cloudDesktopApp.loadApp("desktop",LoadParams);  
cloudDesktopApp.on('stat',function(data){  
    var stat = data;  
});  
cloudDesktopApp.on('ready',function(){  
    console.log('conn ready');  
});  
cloudDesktopApp.on('exit',function(data){  
    $(".datachannel").hide();  
    console.log('exit success');  
});  
  
function getUserSign(reqTime,userKey){
```

```

        var userSign =
hex_md5(hex_md5(userKey+"doliticlouddesktop")+hex_md5(userKey+reqTime));
        return userSign;
    }

```

## 摇杆对应的键配置示例

```

//wasd
var LoadParams = {
    joystickLeftKey : cloudDesktopApp.keyMsgType.KEYCODE_A,
    joystickRightKey : cloudDesktopApp.keyMsgType.KEYCODE_D,
    joystickDownKey : cloudDesktopApp.keyMsgType.KEYCODE_S,
    joystickUpKey : cloudDesktopApp.keyMsgType.KEYCODE_W,
};
//上下左右箭头
var LoadParams = {
    joystickLeftKey : cloudDesktopApp.keyMsgType.KEYCODE_LEFT_ARROW,
    joystickRightKey : cloudDesktopApp.keyMsgType.KEYCODE_RIGHT_ARROW,
    joystickDownKey : cloudDesktopApp.keyMsgType.KEYCODE_DOWN_ARROW,
    joystickUpKey : cloudDesktopApp.keyMsgType.KEYCODE_UP_ARROW,
};

```

## 4 函数列表

### 加载渲染服务

```
cloudDesktopApp.loadApp(domid,params);
```

参数:

**domid** : 指定id的div元素

类型: 字符串

**params**: 服务的初始化参数

类型: json对象

对象参数说明:

**webServerAddr** : 服务地址 【字符串】

**reqTime** : 秒级时间戳, 【字符串】

**userKey** : 请求key:随机字符串, 【字符串】

**userSign** : 请求sign:生成算法 userSign =

Md5(Md5(userKey+"doliticlouddesktop")+Md5(userKey+reqTime)) 【字符串】

**connToken** : 连接token,请通过api或者后台流路列表中创建token, 【字符串】

**connMode** : sdk调用固定为sdk 【字符串】

**runType** : 可选值为app 或者box。运行模式, 如果要支持鼠标无限移动的游戏

类应用可以设置为box 【字符串】

**pointerMode** : 可选值为default, 【字符串】

**moveMode** : 可选值为default, 【字符串】

**touchMode** : 触摸模式, ue类支持触屏的应用可以改为touch 【字符串】

**scaleMode** : 画面比例, 支持height【适应高度】 width【适应宽度】

default【原始尺寸】 native【原始尺寸】 fill【铺满屏幕】 cover【保持比例铺满屏幕自动裁剪】

**loadSkin** : 可选值为default, 【字符串】

**loadType** : 加载模式, 支持管理端web端口 可选: auto stream 以及留空

**group** : 应用识别码, loadType为auto时填写应用识别码

**forceRenderMode** : 可选值为default 【字符串】

**isShowViewOnlyMask**: 是否显示旁边提示的遮罩 【布尔值 true or false】

**isShowSetBar** : 是否显示设置按钮 【布尔值 true or false】

**isShowHandleBar** : 移动端是否显示手柄

**isDefaultShowHandleBar** : 移动端是否默认直接显示手柄, 而不是通过菜单打开. 如果该参数设置为true, 则**isShowHandleBar**强制设置为true

**isForceHorizontal** : 移动端是否强制横屏 【布尔值 true or false】

**isShowKeyBar** : 是否显示键盘按钮 【布尔值 true or false】

**isShowMenu** : 是否默认显示菜单 【布尔值 true or false】

**isShowParam** : 是否默认显示网络参数 【布尔值 true or false】

**isShortcutParam**: 是否启用显示隐藏延迟参数的快捷键 【布尔值 true or false】

**isShowPointer** : /是否显示用户浏览器所在系统的鼠标 【布尔值 true or false】

**isReceiveAudio**: 是否接受音频数据 【布尔值 true or false】

**isAutoExists**: 长时间无操作是否自动退出画面 【布尔值 true or false】

**isShowDefinitionMenu** : 是否显示清晰度列表 【布尔值 true or false】

**isShowScaleMenu** : 是否显示画面比例菜单 【布尔值 true or false】

**isShowResolutionMenu** : 是否显示分辨率列表 【布尔值 true or false】

**isShowShortcutControlMenu** : 是否显示控制按钮 【布尔值 true or false】

**isShowReloadMenu** : 是否显示重启应用按钮 【布尔值 true or false】

**openMicrophone**: false, //开启本地麦克风收集, 必须https方式【布尔值 true or false】

**joystickLeftKey** : 手柄向左滑动模拟的键盘按键【默认cloudDesktopApp.keyMsgType.KEYCODE\_A】

**joystickRightKey** : 手柄向右滑动模拟的键盘按键【默认cloudDesktopApp.keyMsgType.KEYCODE\_D】

**joystickDownKey** : 手柄向下滑动模拟的键盘按键【默认cloudDesktopApp.keyMsgType.KEYCODE\_S】

**joystickUpKey** : 手柄向上滑动模拟的键盘按键【默认cloudDesktopApp.keyMsgType.KEYCODE\_W】

**ignoreKeyboardKeys** : 不向服务器传递的键盘按钮, 具体支持的按键加 键盘常量列表【cloudDesktopApp.keyMsgType的数组】

```
[
    cloudDesktopApp.keyMsgType.KEYCODE_F11
]
```

## 重连服务

cloudDesktopApp.reconnect()

参数:

无。【自动使用第一次loadApp函数的params】

## 退出服务

cloudDesktopApp.exit()

参数:

无。【自动使用第一次loadApp函数的params】

## 获取网络参数

cloudDesktopApp.getAppStat()

参数:

无。

返回值示例:

```
{
  "jitterBufferDelay": 0,
  "framesBufferDelay": 1.21,
  "lossRate": 0,
  "rtt": 0.39,
```

```
"fps": 0,  
"bitrate": 1.86,  
"resolutionWidth": 1920,  
"resolutionHeight": 1080,  
"frameTransDelay": 0.19,  
"endToEndDelay": 11.6,  
"renderDelay": 10  
}
```

返回值参数说明：

jitterBufferDelay: 抖动延迟

framesBufferDelay: 解码缓冲区的延迟

frameTransDelay: 帧传输延迟

renderDelay: 渲染延迟

endToEndDelay: 端到端延迟

lossRate: 丢包率

rtt: 往返时延

fps: 帧率

bitrate: 码率

resolutionWidth: 获取分辨率-宽度

resolutionHeight: 获取分辨率-高度

## 切换当前响应的鼠标方向 【常用于移动端】

```
cloudDesktopApp.setMouseDirection(mouseDirection)
```

参数：

鼠标方向 【int】

可选值：

cloudDesktopApp.mouseMsgType.MOUSE\_CHOOSE\_DIRECTION\_LEFT 鼠标左键

cloudDesktopApp.mouseMsgType.MOUSE\_CHOOSE\_DIRECTION\_MIDDLE 鼠标滚

轮

cloudDesktopApp.mouseMsgType.MOUSE\_CHOOSE\_DIRECTION\_RIGHT 鼠标右

键

## 获取当前响应的鼠标方向 【常用于移动端】

```
cloudDesktopApp.getMouseDirection()
```

参数：

无

返回值：获取当前响应的鼠标方向 【int】

cloudDesktopApp.mouseMsgType.MOUSE\_CHOOSE\_DIRECTION\_LEFT 鼠标左键

cloudDesktopApp.mouseMsgType.MOUSE\_CHOOSE\_DIRECTION\_MIDDLE 鼠标滚轮

cloudDesktopApp.mouseMsgType.MOUSE\_CHOOSE\_DIRECTION\_RIGHT 鼠标右键

## 进入全屏

```
cloudDesktopApp.appEnterFullscreen()
```

参数：

无。

## 退出全屏

```
cloudDesktopApp.appExitFullscreen()
```

参数：  
无。

## 显示网络参数

```
cloudDesktopApp.showParam()
```

参数：  
无。

## 隐藏网络参数

```
cloudDesktopApp.hideParam()
```

参数：  
无。

## 显示菜单

```
cloudDesktopApp.showMenu()
```

参数：  
无。

## 隐藏菜单

```
cloudDesktopApp.hideMenu()
```

参数：  
无。

## 显示键盘

```
cloudDesktopApp.showKeyboard()
```

参数：  
无。

## 隐藏键盘

```
cloudDesktopApp.hideKeyboard()
```

参数：  
无。

## 显示手柄

```
cloudDesktopApp.showJoystick()
```

参数：  
无。

## 隐藏手柄

```
cloudDesktopApp.hideJoystick()
```

参数：  
无。

## 显示加载失败提示

```
cloudDesktopApp.showLoadError(msg)
```

参数：  
msg：加载失败提示

## 显示加载进度提示

```
cloudDesktopApp.showLoadProcess(msg)
```

参数：  
msg：加载进度提示

## 暂停串流

```
cloudDesktopApp.pauseStream()
```

参数：  
无。

## 继续串流

```
cloudDesktopApp.resumeStream()
```

参数：  
无。

## 消息通道发送消息给应用

```
cloudDesktopApp.sendDataChannelMsg(isText,data)
```

参数：  
isText：是否是文本类型  
data：发送的数据

## 重启应用

```
cloudDesktopApp.restartApp()
```

参数：  
无。

## 发送组合键

```
cloudDesktopApp.sendCombinationKey(type)
```

参数：组合键类型 【int】  
0:windows  
1:windows + L  
2:ctrl + alt + delete  
3:ctrl + alt + end

## 发送自定义键值

```
cloudDesktopApp.sendCustomKey(keyCode, isDown)
```

参数：

keyCode：要发送的按键的keyCode只，支持以下9个键值

```
cloudDesktopApp.KEYMSGTYPE.KEYCODE_UP_ARROW,  
cloudDesktopApp.KEYMSGTYPE.KEYCODE_DOWN_ARROW,  
cloudDesktopApp.KEYMSGTYPE.KEYCODE_LEFT_ARROW,  
cloudDesktopApp.KEYMSGTYPE.KEYCODE_RIGHT_ARROW,  
cloudDesktopApp.KEYMSGTYPE.KEYCODE_W,  
cloudDesktopApp.KEYMSGTYPE.KEYCODE_A,  
cloudDesktopApp.KEYMSGTYPE.KEYCODE_S,  
cloudDesktopApp.KEYMSGTYPE.KEYCODE_D,  
cloudDesktopApp.KEYMSGTYPE.KEYCODE_SPACE,
```

isDown：是否按下。按下按键设置为true,释放设置为false

## 5 回调事件

### ready事件：渲染服务加载成功

```
cloudDesktopApp.on('ready',function(){  
    console.log('conn ready');  
});
```

### exit事件：渲染服务退出

```
cloudDesktopApp.on('exit',function(data){  
    console.log('exit success');  
});
```

### stat事件：实时获取网络参数

```
cloudDesktopApp.on('stat',function(data){  
    //实时获取网络参数  
    //获取抖动延迟  
    let jitterBufferDelay = data.jitterBufferDelay;  
    //获取解码缓冲区的延迟  
    let framesBufferDelay = data.framesBufferDelay;  
    //获取帧传输延迟  
    let frameTransDelay = data.frameTransDelay;  
    //获取渲染延迟  
    let renderDelay = data.renderDelay;  
    //获取端到端延迟  
    let endToEndDelay = data.endToEndDelay;  
    //获取丢包率  
    let lossRate = data.lossRate;  
    //获取往返时延  
    let rtt = data.rtt;  
    //获取帧率  
    let fps = data.fps;  
    //获取码率  
    let bitrate = data.bitrate;  
    //获取分辨率-宽度  
    let resolutionwidth = data.resolutionwidth;  
    //获取分辨率-高度
```



```
    let resolutionHeight = data.resolutionHeight;
  });
```

### streamPause事件：串流暂停

```
cloudDesktopApp.on('streamPause',function(){
    console.log('streamPause');
});
```

### streamResume事件：串流继续

```
cloudDesktopApp.on('streamResume',function(){
    console.log('streamResume');
});
```

### appClosed事件：应用关闭

```
cloudDesktopApp.on('appClosed',function(){
    console.log('appClosed');
});
```

### clientKilled事件：客户端被退出

```
cloudDesktopApp.on('clientKilled',function(){
    console.log('clientKilled');
});
```

### dataChannelConnected事件：消息通信连接成功

```
cloudDesktopApp.on('dataChannelConnected',function(){
    console.log('dataChannelConnected');
});
```

### dataChannelDisconnected事件：消息通信断开连接

```
cloudDesktopApp.on('dataChannelDisconnected',function(){
    console.log('dataChannelDisconnected');
});
```

### dataChannel事件：实时消息发送

```
cloudDesktopApp.on('dataChannel',function(data){
    console.log(data);
});
```

## 6 键盘常量列表

```
KEYCODE_BACKSPACE : 8,
KEYCODE_TAB       : 9,
KEYCODE_ENTER     : 13,
KEYCODE_SHIFT     : 16,
```

KEYCODE\_CTRL : 17,  
KEYCODE\_ALT : 18,  
KEYCODE\_PAUSE\_BREAK : 19,  
KEYCODE\_CAPS\_LOCK : 20,  
KEYCODE\_ESCAPE : 27,  
KEYCODE\_SPACE : 32,  
KEYCODE\_PAGE\_UP : 33,  
KEYCODE\_PAGE\_DOWN : 34,  
KEYCODE\_END : 35,  
KEYCODE\_HOME : 36,  
KEYCODE\_LEFT\_ARROW : 37,  
KEYCODE\_UP\_ARROW : 38,  
KEYCODE\_RIGHT\_ARROW : 39,  
KEYCODE\_DOWN\_ARROW : 40,  
KEYCODE\_INSERT : 45,  
KEYCODE\_DELETE : 46,  
KEYCODE\_F1 : 112,  
KEYCODE\_F2 : 113,  
KEYCODE\_F3 : 114,  
KEYCODE\_F4 : 115,  
KEYCODE\_F5 : 116,  
KEYCODE\_F6 : 117,  
KEYCODE\_F7 : 118,  
KEYCODE\_F8 : 119,  
KEYCODE\_F9 : 120,  
KEYCODE\_F10 : 121,  
KEYCODE\_F11 : 122,  
KEYCODE\_F12 : 123,  
KEYCODE\_0 : 48,  
KEYCODE\_1 : 49,  
KEYCODE\_2 : 50,  
KEYCODE\_3 : 51,  
KEYCODE\_4 : 52,  
KEYCODE\_5 : 53,  
KEYCODE\_6 : 54,  
KEYCODE\_7 : 55,  
KEYCODE\_8 : 56,  
KEYCODE\_9 : 57,  
KEYCODE\_A : 65,  
KEYCODE\_B : 66,  
KEYCODE\_C : 67,  
KEYCODE\_D : 68,  
KEYCODE\_E : 69,  
KEYCODE\_F : 70,  
KEYCODE\_G : 71,  
KEYCODE\_H : 72,  
KEYCODE\_I : 73,  
KEYCODE\_J : 74,  
KEYCODE\_K : 75,  
KEYCODE\_L : 76,  
KEYCODE\_M : 77,  
KEYCODE\_N : 78,  
KEYCODE\_O : 79,  
KEYCODE\_P : 80,  
KEYCODE\_Q : 81,  
KEYCODE\_R : 82,  
KEYCODE\_S : 83,  
KEYCODE\_T : 84,

KEYCODE\_U : 85,  
KEYCODE\_V : 86,  
KEYCODE\_W : 87,  
KEYCODE\_X : 88,  
KEYCODE\_Y : 89,  
KEYCODE\_Z : 90,  
KEYCODE\_SUB : 189,  
KEYCODE\_EQUAL : 187,  
KEYCODE\_LEFT\_BRACKET : 219,  
KEYCODE\_RIGHT\_BRACKET : 221,  
KEYCODE\_Y\_AXIS : 220,  
KEYCODE\_SEMICOLON : 186,  
KEYCODE\_UP\_COMMA : 222,  
KEYCODE\_DOWN\_COMMA : 188,  
KEYCODE\_POINT : 190,  
KEYCODE\_QUESTION : 191,  
KEYCODE\_BACK\_QUOTE : 192,  
KEYCODE\_NUM\_0 : 96,  
KEYCODE\_NUM\_1 : 97,  
KEYCODE\_NUM\_2 : 98,  
KEYCODE\_NUM\_3 : 99,  
KEYCODE\_NUM\_4 : 100,  
KEYCODE\_NUM\_5 : 101,  
KEYCODE\_NUM\_6 : 102,  
KEYCODE\_NUM\_7 : 103,  
KEYCODE\_NUM\_8 : 104,  
KEYCODE\_NUM\_9 : 105,  
KEYCODE\_NUM\_MUL : 106,  
KEYCODE\_NUM\_ADD : 107,  
KEYCODE\_NUM\_ENTER : 108,  
KEYCODE\_NUM\_SUB : 109,  
KEYCODE\_NUM\_DOT : 110,  
KEYCODE\_NUM\_DIV : 111,  
KEYCODE\_NUM\_LOCK : 144,